

NCF-74-107
DRL No. T-815
DRL Line Item 2
DPO MA-129T

NASA CR-134391

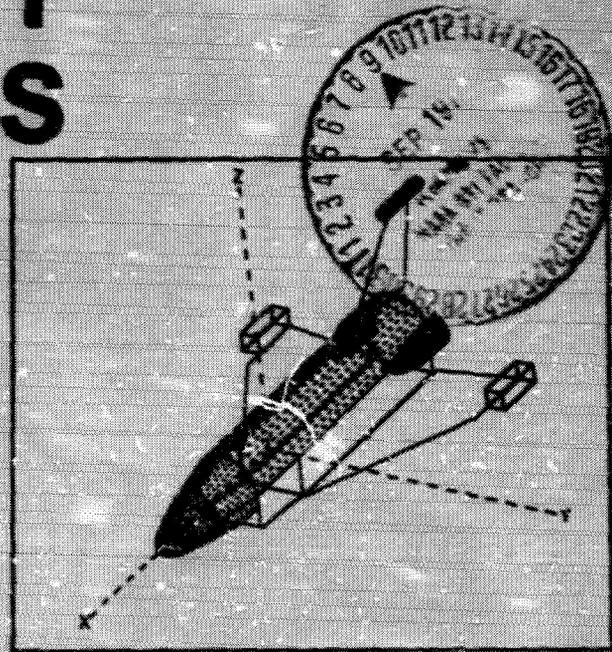
Final
Report

June 1974

Thermal Radiation Analysis System

T R A S Y S

(NASA-CR-134391) THERMAL RADIATION
ANALYSIS SYSTEM (TRASYS) Final Report
(Martin Marietta Aerospace, Denver, Colo.)
59 p HC 46.00
G3/33 (Inclan
46865



MARTIN MARIETTA

MCR-74-100

Contract NAS9-13033
DRL Number T-815
DRL Line Item 2
DRD Number MA-129T

Final
Report

June 1974

THERMAL RADIATION
ANALYSIS SYSTEM

T
R
A
S
Y
S

Prepared for

National Aeronautics and Space
Administration
Lindon B. Johnson Spacecraft Center
Contract NAS9-13033

Prepared by


Carl L. Jensen


Richard G. Goble

MARTIN MARIETTA CORPORATION
P. O. Box 179
Denver, Colorado 80201

FOREWORD

The Martin Marietta Thermal Radiation Analyzer System (TRASYS) program marks the first instance that thermal radiation analysis has been put on the same basis as thermal analysis using program systems such as MITAS and SINDA. As with these thermal analyzer programs, the user is provided the powerful options of writing his own executive or driver logic and choosing, among several available options, the most desirable solution technique(s) for the problem at hand. In addition, many features never before available in a single radiation analysis program are provided. Among the more important are:

- 1000 node problem size capability with shadowing by intervening opaque or semi-transparent surfaces;
- choice of diffuse, specular or diffuse/specular radiant interchange solutions;
- capability for time variant geometry in orbit;
- choice of analytically determined or externally supplied shadow data for environmental flux calculations;
- form factors and environmental fluxes computed using an internally-optimized number of surface grid elements, selected on the basis of user-supplied accuracy criteria;
- a general edit capability for updating thermal radiation model data stored on tape;
- a plot package that provides a pictorial representation of the user's geometry.

TRASYS is indebted to a number of predecessor programs in the thermal radiation analysis field. The major contributors were HEATRATE, MTRAP Version 2.0, RADFAC and the MRI Computer Program for Determining External Radiation Absorbed by the Apollo Spacecraft.

This report is a documentation of the TRASYS development conducted over the time period between July 1972 and April 1974.

This report was generated under NASA Contract NAS9-13033, "Development of a Thermal Radiation Analysis/Heat Rate Computer Program System." The technical monitoring for this program was provided by Mr. Robert A. Vogt of the Thermal Technology Branch of the Structures and Mechanics Division, NASA Lyndon B. Johnson Space Center.

CONTENTS

		<u>Page</u>
I.	INTRODUCTION	1
II.	MTRAP 1.0 CONVERSION AND DELIVERY	3
III.	EXISTING PROGRAM EVALUATION	5
A.	VIP Program	5
B.	EANCAL Program	7
C.	MRI Program	7
D.	SINDA Program	8
E.	TRIP and RADFAC Programs	8
IV.	PROGRAM DESIGN	11
A.	System Structure	11
B.	Preprocessor	11
C.	Processor	11
V.	PROGRAM INCREMENT 1 DEVELOPMENT	15
A.	Preprocessor Development	15
B.	Form Factor Segment	17
C.	Node Plotter Segment	21
D.	Orbit Plotter Segment	22
E.	Radiation Interchange Segments	23
F.	Incident and Absorbed Radiation Segments	24
VI.	PROGRAM INCREMENT 2 DEVELOPMENT	33
A.	Specular-Diffuse Radiation Interchange	33
B.	Output Data Plotter Segment	38
C.	Trajectory Tape Capability	38
VII.	PROGRAM INCREMENT 3 DEVELOPMENT	39
A.	Modification of Large Blocks of Surface Data	39
B.	Added Surface Data Input Options	39
C.	Automatic Node Plots	40
D.	Automatic Orbit Generation	40
E.	Shadower Surfaces Only	40
F.	Radiation Network Simplification	41
G.	Equivalent Form Factors	47
H.	Processor Library Improvements	48
I.	Direct Irradiation with Specular-Diffuse Surfaces	50
J.	Conversion to IBM 370	51
VIII.	PROGRAM EVALUATION AND OPTIMIZATION	53 and 54

Figure

IV-1	TRASYS Macro Flow Chart	12
V-1	Determination of Form Factors	18
V-2	Error Characteristics for Identical, Parallel, Directly Opposed Rectangles	19
V-3	Determination of Direct Incident Solar Flux	25
V-4	Determination of Node-to-Planet Form Factors	27
V-5	Planet Element Breakdown	28
VII-1	MESS Technique One-Way Conductors	45

I. INTRODUCTION

The Thermal Radiation Analysis System, TRASYS, is a digital computer software system with generalized capability to solve the radiation-related aspects of thermal analysis problems.

When used in conjunction with a generalized thermal analyzer program such as the Systems Improved Numerical Differencing Analyzer (SINDA) program, any thermal problem that can be expressed in terms of a lumped parameter R-C thermal network can be solved. The function of TRASYS is twofold. It provides:

- a. Internode radiation interchange data; and
- b. Incident and absorbed heat rate data from environmental radiant heat sources.

Data of both types is provided in a format directly usable by the thermal analyzer programs. One of the primary features of TRASYS is that it allows the user to write his own executive or driver program which organizes and directs the program library routines toward solution of each specific problem in the most expeditious manner. The user also may write his own output routines, thus the system data output can directly interface with any thermal analyzer using the R-C network concept.

Other features of TRASYS include:

- a. 1000 node allowable problem size;
- b. time variable problem geometry allowed;
- c. edit capability allowing the combination or separation of multiple thermal radiation models.
- d. a plot package that provides pictorial plots of input geometry and orbit data as well as output data.

The TRASYS system consists of two major components: (1) the preprocessor, and (2) the processor library. The preprocessor has two major functions. First, it reads and converts the user's geometry input data into the form used by the processor library routines. Second, it accepts the user's driving logic written in the TRASYS modified FORTRAN language that directs user-provided and/or library routines in the solution of the problem. The processor library consists of FORTRAN language routines that perform

the functions needed by the user. The user has, in some cases, a choice of solution techniques to perform the same function.

TRASYS was developed and delivered in three increments. These three increments, plus program evaluation and optimization comprise the contract statement of work task list. The descriptive material in this report will address these tasks individually.

II. MTRAP 1.0 CONVERSION AND DELIVERY

During July and August of 1972, the thermal radiation analyzer program that has been operational and in use at Martin Marietta since 1970 (Reference 1) was delivered to NASA-MSD and made operational on the UNIVAC 1108 (Exec II operating system). This delivery involved the completion of the following subtasks:

- a. Code was generated to translate plot routine commands from the CDC 280 plot package language to that of the Univac 1108 SC4060 plot package language.
- b. Program code was changed to make the program compatible with the 36 bit Univac word. (The CDC computer utilizes a 60 bit word.)
- c. Program code changes to reflect FORTRAN language differences between the Univac and CDC computers.
- d. Program code changes to reflect the different printer line lengths (136 characters on CDC, 152 on Univac).
- e. Program dimension changes to fit the largest possible problem into the Univac core. Initially, the original 140 node capability was not achieved because of word length difference and differences in approach to core loading. The initial Univac version had 125 node capability.

Effort continued during the first six months of the contract period in order to provide 140 node capability. The problem was finally resolved by splitting the direct flux computation segment into two parts and coding a communication link between them. This provided sufficient core to compile the direct flux code with the program dimensioned for 140 nodes.

During this period, MTRAP 1 was converted to the EXEC VIII operating system as a zero cost delivery. This provided NASA-JSC thermal analysis personnel with an extra computer access.

III. EXISTING PROGRAM EVALUATION

A. VIP PROGRAM

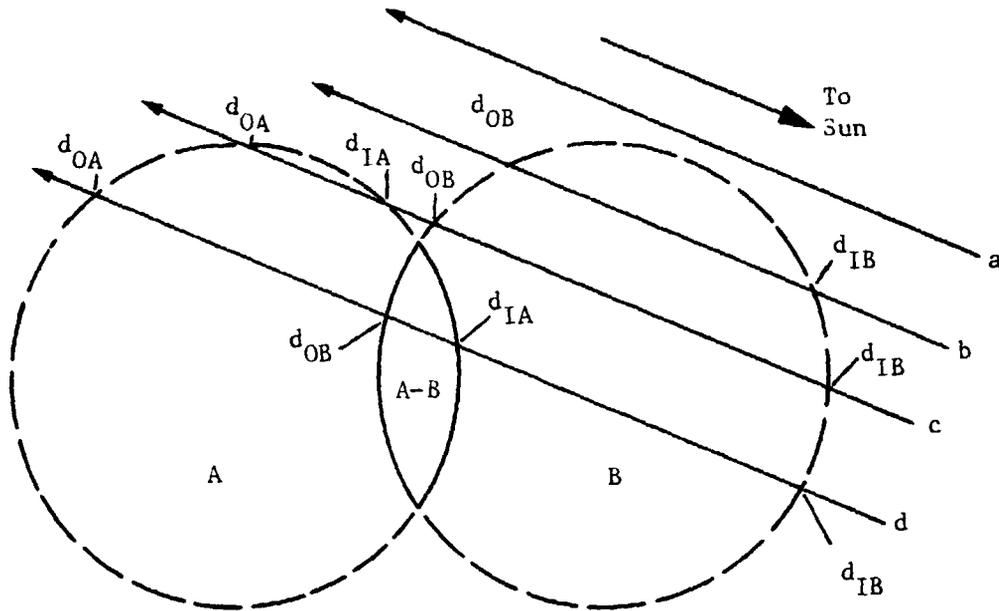
A source deck and documentation for the Vehicle Illumination Program "VIP" were obtained. This program calculates orbital environmental incident heat, and employs a unique method of arbitrarily combining a number of different geometric shapes in order to describe spacecraft geometry. Evaluation of this program was directed at determining the methods used in the geometry combination technique and to determine if incorporation of these methods into TRASYS was desirable and feasible.

This evaluation was performed and the following conclusions can be stated:

1. The VIP program utilizes complete, closed volumes rather than surfaces as its basic geometric building blocks. This is an adequate means of geometrically describing a spacecraft but an inadequate means of describing a spacecraft for thermal modeling. For instance, a right circular cylinder would rarely appear in a thermal model as a complete entity but rather as eight or more separate nodal surfaces. In comparison, the basic geometries used in the Martin Marietta and Lockheed thermal radiation analyzer programs are much more useful.
2. The geometric combination technique is clever and efficient. The addition and subtraction of volumes, however, results in other closed volumes, not surfaces, that still suffer from the inability to simulate required nodal breakdown. The method is illustrated as follows:

Consider the arbitrary lens shape obtained by the subtractive combination of two spheres, input as A minus B. Computation of the solar energy incident on the lens is illustrated.

¹R. Goldstein and R. Nagel: *Computer Simulation of Thermal Radiation in Spacecraft*, Mathematical Applications Group, Inc., White Plains, NY, March 1969.



Rays a, b, c and d are members of an arbitrarily large set of lines, each representing a quantum of solar flux. The rays are "fired" from random locations on a disc just large enough to completely shadow the spacecraft when oriented normal to the insolation vector. The distances from this disc to the points where each ray enters and leaves each geometric component (spheres A and B) are computed. Knowing these distances allows computation of the solar flux on A and B by applying the following logic:

- a. Type a rays miss both spheres and are ignored.
- b. Type b rays miss sphere A and are ignored.
- c. Type c rays are ignored because $d_{IA} > d_{OB}$.
- d. Type d rays are "counters" because $d_{OB} > d_{IA}$.

It is apparent that applying this logic to a sufficient number of rays will result in a sufficiently accurate value for the solar flux on the lens A - B. Note that a simple change in the inequality logic applied would result in computing the flux on the double-sphere shape A + B.

It was concluded from this investigation that the VIP geometry combining method was not applicable to TRASYS primarily because it is based upon ray tracing. Ray tracing represented a major departure from flux computation methods planned for the development program. A change to ray tracing would have been impossible within the available time and money resources.

B. EANCAL PROGRAM

The EANCAL Program was developed as a support tool for people using the HEATRATE, LOHARP¹ or MTRAP² programs. It performs the function of transforming surface descriptions that are in terms of Cartesian coordinates into descriptions compatible with HEATRATE, etc. This eliminates the sometimes laborious and error prone process of computing the rotation and translation parameters necessary to locate surfaces in relation to the central coordinate system of the problem.

Experience with the EANCAL concept indicated there was no question of the desirability of incorporating this capability into TRASYS. This was done by incorporating the transformation code into the TRASYS surface data processing routines. It was necessary to expand EANCAL capability so that polygons and boxes (not available in HEATRATE, etc.) are allowed. In general, the EANCAL code was not retained per se, but rather the same mathematical methods were utilized in the TRASYS routines.

C. MRI PROGRAM

The Midwest Research Institute Program for Determining External Radiation Absorbed by the Apollo Spacecraft³ has been in the Martin Marietta Aerospace program library for some time and has been used enough to allow assessment of its capability. The MRI program's primary strong point is its ability to compute absorbed fluxes in minimum computer time. This is accomplished at the expense of a relatively crude representation of the problem:

¹Lockheed Orbital Heat Rate Package, LOHARP, Lockheed Missiles and Space Company.

²G. M. Holmstead and Ronald E. Paulson: *Martin Marietta Thermal Radiation Analyzer Program (MTRAP)*. M-70-72. Martin Marietta Corporation, Denver, Colorado, September 1970.

³*Development of a Computer Program for Determining External Radiation Absorbed by the Apollo Spacecraft*, Midwest Research Institute.

all surfaces are assumed planar, energy reflections are ignored, and all information describing the internode blockage (shadowing) must be obtained externally and supplied as a shadow tape. Generating shadow tapes led to a very laborious and long lead time process involving a scale model of the spacecraft and a collimated light beam. This effectively limited the MRI program to no-blockage configurations or to long-lived configurations in well-funded programs.

Another MRI program strong point is that it allows the user the choice of defining his orbit in standard orbital mechanics terms, or in a convenient orbit-referenced coordinate system.

The shadow tape capability and orbit definition flexibility of the MRI program were both incorporated into TRASYS. The identical packed shadow tape format was used as were the transform routines involved in defining sun/planet/spacecraft position relationships from orbit definition. From the user's standpoint, the orbit definition inputs for TRASYS are essentially identical to those of the MRI program.

The shadow tape capability, which included in TRASYS the capability to analytically generate shadow tapes, represents a real advance in the art of computing orbital heat fluxes. The primary drawbacks of the MRI program are eliminated and the use of the shadow tape for shadowing computations is so rapid that some problems require less processor time, even including shadow tape generation, than if the shadowed fluxes were all computed analytically.

D. SINDA PROGRAM

At the suggestion of the NASA-JSC contract monitor, the NEWMOD edit segment of the SINDA thermal analyzer was examined for applicability to TRASYS. It was determined that NEWMOD provided excellent features in the area of combining existing thermal models as well as convenient editing. It was decided to incorporate the NEWMOD features without actually integrating SINDA code into TRASYS.

E. TRIP AND RADFAC PROGRAMS

The TRIP¹ and RADFAC² programs were evaluated for applicability

¹Direct View Factor Program and Generalized Radiative Heat Transfer Program, TRIP, The Boeing Company.

²A Computer Program for Determination of Radiation Interchange Factors, RADFAC, Hughes Aircraft Company

in providing radiation interchange solutions for real body surfaces. (Emphasis on specular and specular-diffuse surfaces.) The TRIP program, which uses a ray tracing technique, was rejected because ray tracing is too great a departure from the basic double-integration method planned for TRASYS to allow it to be reasonably integrated.

RADFAC was much more compatible because it generates radiation interchange factors somewhat analogous to diffuse gray-body factors and also begins with the form factor concept. The most serious limitation of RADFAC is its lack of a general capability to compute shadowed form factors. For this reason, it was decided to use the diffuse double integration form factor routines from MTRAP to obtain the shadowed form factors and use new MTRAP based routines to compute the surface images and image factors required in the RADFAC approach. The final interchange factors are computed using equations resulting from an extension of the Gebhardt method.

IV. PROGRAM DESIGN

A. SYSTEM STRUCTURE

The system structural design settled upon for TRASYS is quite similar to the dual execution used in the CINDA family of thermal analyzer programs. The primary difference between TRASYS and these programs is the overlay or segmentation approach used. Only enough code and data is in core at one time to solve a particular portion of the overall problem. The first execution reads and processes the input data, performing edit operations where called for, and writes the FORTRAN program that constitutes the user's driver logic. The processed and compressed data is passed on to the processor and the driver program via the FORTRAN compiler, whereupon the processor segments are loaded and executed. Only the portions of the processor library necessary for the problem at hand are loaded. Figure IV-1 is a macro flow chart of the system.

B. PREPROCESSOR

The functions assigned to the preprocessor execution are as follows:

- a. Collect input data from tape and/or card sources, edit as necessary and write binary EDITO tape.
- b. Read and interpret all data input, check for errors, and make all one-time only calculations necessary to compress and organize data for use by the processor; write processor data files.
- c. Read and interpret user logic input, write FORTRAN program ODPROG from this.
- d. Read in and organize processor generated restart data.

A preprocessor segment is assigned to each of these functions. At this time, the restart segment is skeletal only. Resources did not permit developing an optimum restart capability. Restart is accomplished using program generated BCD data input through the normal data input blocks.

C. PROCESSOR

The processor assembles differently depending on the problem at hand. A problem using all the capabilities of TRASYS would assemble a processor consisting of the user's driver logic program (ODPROG) plus all of the following program segments:

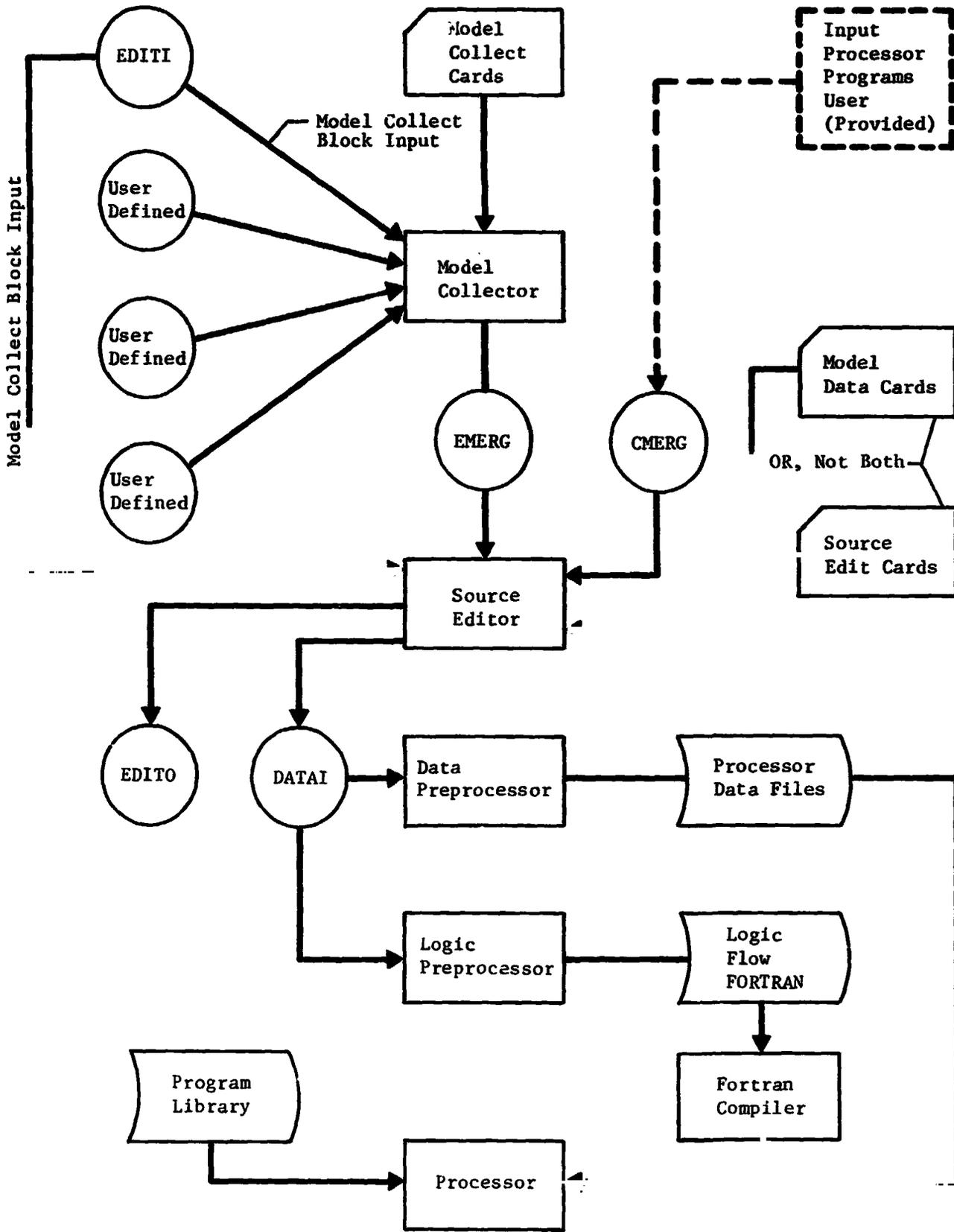


Figure IV-1 TRASYS Macro Flow Chart

- a. FFCAL (computes form factors).
- b. RBCAL (computes radiant interchange factors when specular-diffuse surfaces are input. Called only from FFCAL).
- c. GBCAL (computes radiant interchange factors).
- d. DICAL (computes direct irradiation).
- e. DRCAL (computes components of direct solar flux reflected from specular-diffuse surfaces. Called from DICAL only).
- f. AQCAL (computes absorbed heat rates).
- g. RKCAL (converts grey body factor matrix into radiation conductors in thermal analyzer input format).
- h. QOCAL (converts absorbed heat data into energy vs time and/or orbital average energy tables in thermal analyzer input format).
- i. NPLOT (provides pictorial plots of user's surface data input).
- j. OPLOT (provides pictorial plots of user's surface data as it relates to solar and planet heat sources).
- k. PLOT (provides x vs y plots of output data).
- l. SFCAL (computes shadow factor tables for shadow tapes).
- m. RCCAL (condenses the radiation interchange data from large problems; outputs radiation conductors).

The FFCAL, DICAL, GBCAL and AQCAL segments generate and work with such voluminous quantities of data that an elaborate out-of-core data storage scheme is required. This data is written and retrieved according to time, orbit true anomaly or an ODPROG sequence numbering scheme called steps. Figure IV-2 illustrates this data storage scheme.

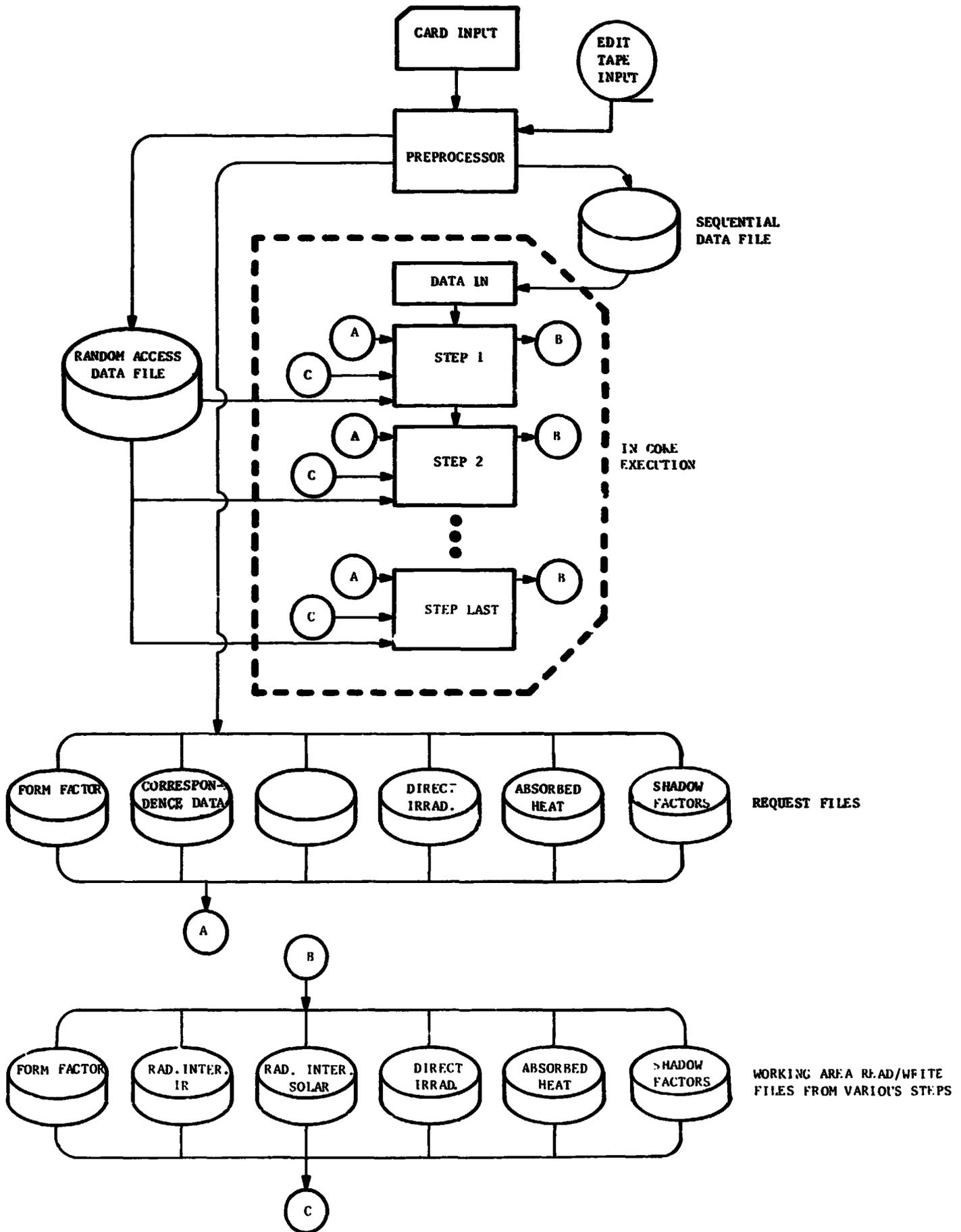


Figure IV-2 Program Data Storage Scheme

V. PROGRAM INCREMENT 1 DEVELOPMENT

The functional definition of program increment 1 was a radiation analysis program with the same compute/output capabilities as MTRAP 1, but with the increased capabilities of TRASYS in the areas of data input options, edit capability, increased problem size, and movable geometry. This development process was accomplished over the first 10 months of the contract performance period, culminating in delivery to the JSC Univac computer facility in April 1973. During this period, the decision was made to install the program on the Univac EXEC VIII operating system at JSC rather than the EXEC II system specified in the Statement of Work. This enabled the programmers to take advantage of the advanced features of EXEC VIII for efficiency in core utilization. Also, the DEMAND feature of EXEC VIII made it possible to set up a remote terminal in Denver for use in transmitting new code and program "fixes" directly to the permanent files in Houston.

A. PREPROCESSOR DEVELOPMENT

Since no existing radiation analysis program had the data input capabilities and the provision for the user to write driver logic that was envisioned for TRASYS, it was necessary to write approximately 80 percent of the preprocessor code in its entirety. The remaining 20 percent came from existing MTRAP routines that compute the required transformation matrices and other one-time computations on the input data. In some cases these MTRAP routines needed extensive modifications in order to eliminate inconsistencies in the definition of coordinate system rotations. The preprocessor coding was done by programming personnel having extensive experience with a preprocessor with a somewhat similar function in the MITAS thermal analyzer programs. Features and programming techniques from the MITAS preprocessor were used, rather than blocks of code per se. Similarly, the techniques used in the SINDA editor routines were incorporated in the editor segment.

The major portion of the preprocessor is concerned with numeric input and Hollerith flags. This portion may be referred to as the data preprocessor and in increment 1 it read and processed the following blocks of numeric and Hollerith data:

- a. Options Data
- b. Edit Data
- c. Quantities Data
- d. Array Data
- e. Surface Data
- f. BCS Data
- g. Form Factor Data
- h. Flux Data
- i. Correspondence Data

The input is blocked under header cards with the titles above. Each card in the block is read and analyzed column by column. The basic card input format is variable length fields, separated by commas. The variables input are not order dependent except when unavoidable. Input formats take two basic forms. Single variables, real, integer or Hollerith may be input according to:

NAME = DV

Multiple variables, where allowed, take the form:

NAME = DV1, DV2 ---- DVN

for numeric data, and the special form

NAME = *I AM A HOLLERITH ARRAY*

for Hollerith data longer than 6 letters. Fortran arithmetic expressions, excluding functions, may be used for any numeric data value.

The routines for reading and processing the surface data comprise the largest single part of the data preprocessor. The surface data is analyzed extensively as read and error messages generated to aid in input debugging. The errors found are not fatal to preprocessor execution. A flag is set, however, to prevent passing incorrect input data on to the computation segments of the processor. The array input option is used in the surface data to allow shorthand input of two or more related numbers under one name. For instance, up to four surface properties, the solar and infrared diffuse absorptivities and the solar and infrared transmissivities may be entered according to:

PROP = DV1, ---- DVN

where N = 2 or 4.

Specular properties were not allowed in program increment 1, but allowance for them was made in the preprocessor.

The flux and form factor data blocks may be used to re-input previously computed data, thus providing restart for runs interrupted for any reason. This is an inefficient method of restart because there should be no need of a column by column analysis of machine-generated BCD data that can be expected to be strictly ordered and formatted. This has been identified as an area for further development.

The preprocessor routines that read and process the operations data and subroutine data blocks may be defined as the instructional preprocessor. The operations data block consists of FORTRAN statements plus special statements of the form:

L SEGNAME

and

STEP DV.

The FORTRAN statements, plus any L and STEP cards are used by the instructional preprocessor to write program ODPROG, a FORTRAN program which serves to direct the solution of the user's problem. FORTRAN statements in the operations data are transferred verbatim to ODPROG. L-cards result in generating the code necessary to execute program segment SEGNAME. STEP cards set integer values that enable the user to conveniently refer to and retrieve previously computed data in out of core storage.

The subroutine data block consists of valid and complete FORTRAN routines that the user provides for use in his execution. The instructional processor functions to read these routines, provide the necessary labeled common blocks, pass them to the compiler and load them along with the processor library routines. Note that the FORTRAN compiler does the major portion of input error checking within the operations and subroutine data blocks.

B. FORM FACTOR SEGMENT

The form factor computation segment writes out a matrix of form factors for a given geometry, either through computation or through passing on values found in the form factor data block. This segment required the least development because it basically existed, in the form required for program increment 1, as a part of the MTRAP 2 prototype program already in use. The main

departure from previous practice (MTRAP 1, LOHARP) found in TRASYs form factor methods is the automatic elemental grid sizing according to user input accuracy criteria. This logic is explained in the following section.

1. Elemental Grid Sizing with No Shadowing

The form factor for two finite areas, A_I and A_J (Figure V-1), is defined as

$$F_{IJ} = \frac{1}{A_I} \int_{A_I} \int_{A_J} \frac{\cos \theta_i \cos \theta_j}{\pi r_{ij}^2} dA_J dA_I. \quad (V-1)$$

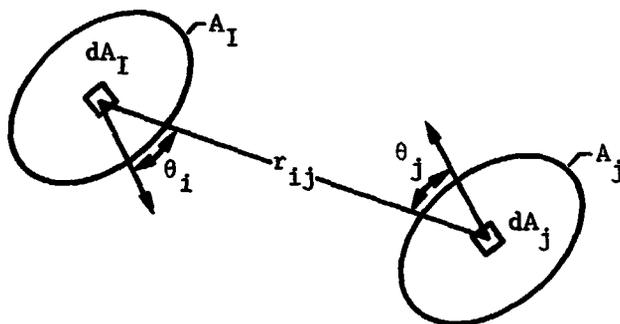


Figure V-1 Determination of Form Factors

A finite difference approximation of Equation V-1 is

$$F_{IJ} = \frac{1}{A_I} \sum_{i=1}^n \sum_{j=1}^m \frac{\cos \theta_i \cos \theta_j}{\pi r_{ij}^2} A_j A_i. \quad (V-2)$$

Equation V-2 approaches an exact representation of Equation V-1 as the sizes of the elemental areas, A_i and A_j , approach zero. The accuracy of form factors calculated by the use of Equation V-2 is, therefore, a function of the size of the elemental areas, A_i and A_j , as compared to the square of the elemental separation distance, r_{ij} . Considering this, it is convenient to define the dimensionless ratio

$$FFACC = \frac{A_i \cos \theta_i \cos \theta_j}{r_{ij}^2} \quad (V-3)$$

for use in controlling form factor accuracy by properly sizing the elemental areas. Figure V-2 shows the relationship between this dimensionless ratio (FFACC) and the form factor error for identical, parallel, directly opposed rectangles.

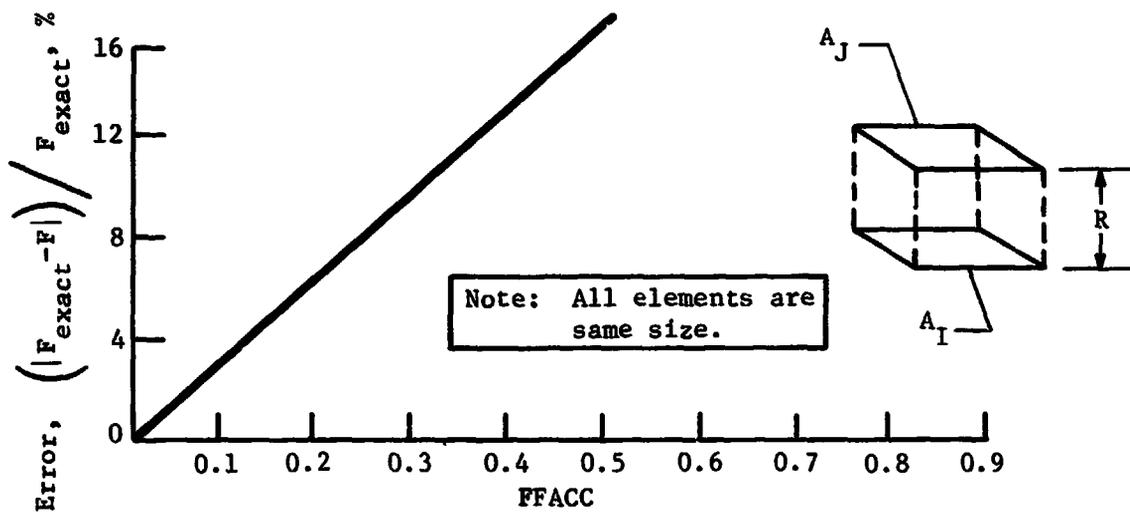


Figure V-2 Error Characteristics for Identical, Parallel, Directly Opposed Rectangles

The TRASYS program employs a technique using Equation V-3 to automatically select the elemental grid sizes for each node pair consistent with the user-defined accuracy parameter, FFACC. (FFACC defaults to 0.05 if it is not defined by the user.)

If all elemental areas on each of two nodes were the same size and had the same separation distance, r_{ij} , the apparent number of elements on a node to satisfy the accuracy parameter, FFACC, would be (considering Equation V-3).

$$N_I = \frac{A_I}{A_i} = \frac{A_I \cos \theta_i \cos \theta_j}{FFACC r_{ij}^2} \quad (V-4)$$

However, since each element pair on the two nodes may have a different separation distance, a different apparent number of equal-sized elements will be indicated for each element pair considered. To take this into account, TRASYS uses a simple arithmetic average of the numbers of elements indicated by the various

element pairs. That is,

$$N_{OPT_{IJ}} = \frac{A_I}{FFACC_{m_i m_j}} \sum_{i=1}^{m_i} \sum_{j=1}^{m_j} \frac{\cos \theta_i \cos \theta_j}{r_{ij}^2} \quad (V-5)$$

where m_i and m_j are initial numbers of elements arbitrarily assigned to obtain a representative mapping of nodes I and J. (m_i and $m_j = 15$ elements in TRASYS).

The total number of elements defined by Equation V-5 is distributed uniformly over the node using a criterion that attempts to make the elements square. The arithmetic average technique assumes that the mean separation distance between nodes is large compared with the variation of separation distance over the two nodes. A check is made to see if this assumption is violated. The maximum number of elements defined by any element pair on the two nodes (Equation V-4) is compared with the arithmetic average value (Equation V-5). If the ratio of N_{max}/N_{opt} is greater than a user-defined value, FFRATL, the two nodes are temporarily divided into subnodes. (FFRATL defaults to 15.0 if it is not input by the user.) The numbers of subnodes used are proportional to $(N_{max}/N_{opt})_I$ and $(N_{max}/N_{opt})_J$. The optimum grid elements are computed independently for each subnode using a separation-distance, weighted-average criterion rather than an arithmetic one. The form factors resulting from the subnode pairs are then combined using form factor algebra. Thus, elemental grids vary for each form factor and may be nonuniform over a node as required to satisfy input accuracy requirements.

2. Elemental Grid Sizing Considering Shadowing

In the detailed element-to-element shadowing checks, an element pair is either completely shadowed or not at all. The accuracy, then, of representing the shadow is proportional to the total number of elements on both nodes. The number of elements on the shadowing surface(s) does not enter into the problem.

In order to conserve computer time, it is assumed that accurate shadowing is required only for relatively large form factors. Therefore, the number of elements required for accurate shadowing is considered to be proportional to the unshadowed form factor, F_{IJ} . That is,

$$N_{S_I} = \frac{F_{IJ} B}{FFACCS}$$

$$N_{S_J} = \frac{F_{JI} B}{FFACCS}$$
(V-6)

where,

F_{IJ} and F_{JI} are computed using Equation V-2 with $n = m_i$ and $m = m_j$, the arbitrarily assigned numbers of elements, as described above.

FFACCS is a user-defined shadowing accuracy factor (defaults to 0.1 if not input).

B is a constant of proportionality (assigned a value of 10 on the basis of experience in trading off computer time vs accuracy).

The number of elements used on any given node for form factors is taken as the maximum of those defined by Equations V-5 and V-6.

TRASYS limits the number of elements on any given node to a maximum of 500 and a minimum of 3.

C. NODE PLOTTER SEGMENT

This segment's purpose is for checking the user's surface data inputs to be certain that the problem geometry is truly as desired. It generates pictorial views of the problem geometry as seen from 3 directions, plus a 3-D and a general view.

The surface plotter logic in MTRAP 1 was used as the foundation to build the node plotter. Surfaces may be defined as portions of a given geometric surface (cylinder, disc, etc) that may be divided into any number of nodes. A surface plot, as in MTRAP 1, provides a picture of the geometry but not of the nodal breakdown. It was decided by mutual agreement with the JSC technical monitor that more informative, albeit cluttered, node plots would be preferable; thus, a node plotter that "draws" all nodes was developed.

Logic was developed to calculate transformation matrices for four built-in views as well as any general view of the problem geometry. The built-in views depict the problem as seen from the x-axis, y-axis, z-axis and a 3-D view. A general view results when the user defines the Euler angles required to rotate a reference

coordinate system into the coordinate system of the view he desires. Node plotter output is controlled by an optional subroutine call prior to execution of the plot segment. Arguments in this subroutine allow the user to define his view(s) desired, scale the plot as desired, define a plot frame title and input the rotation data for a general view. If this subroutine call is not made, the plot segment produces 4 standard views, automatically scaled, under the problem title. The node plotter also has a selective plotting feature. If the user desires only a portion of the nodes in his geometry to be plotted, he may input an array of node numbers specifically defining the nodes desired on the plots. Up to six plot definitions (subroutine calls) may be made prior to node plotter execution, thereby eliminating the time required to load the segment repeatedly.

D. ORBIT PLOTTER SEGMENT

This segment was developed for the purpose of checking user-defined orbital parameters and vehicle orientation prior to calculation of direct incident environmental fluxes.

Logic was developed to calculate transformation matrices for four built-in views, as well as any general view, taking orbital parameters, true anomaly, vehicle orientation, and vehicle spin into account. One orbit position is plotted per frame showing the vehicle and its relationship to the planet and/or the sun. The four built-in views are:

1. 3-D, a three-dimensional plot of the vehicle and the planet.
2. CIGMA, a plot with the orbital plane in the plane of the paper as viewed from the north pole of the orbital plane. This plot shows true views of true anomaly and CIGMA (the angle measured from periapsis to the projection of the solar vector on the orbital plane).
3. BETA, an edge-on view of the orbital plane with the solar vector in the plane of the paper. This plot shows a true view of BETA (the angle measured from the north pole of the orbital plane to the solar vector).
4. SUN, a plot of the vehicle and the planet as viewed from the sun.

The user may also specify any general view he desires by defining the Euler angles necessary to rotate a reference coordinate system into the coordinate system of the view he would like to see plotted.

The input philosophy was made to parallel that used in the node plotter segment as closely as possible. This allows up to six plot definitions to be made prior to a call to the orbit plotter segment, thereby significantly reducing the time required for segment loading. No input prior to the segment call results in a default to plot the four built-in views at the c anomaly that exists in core at the time of the segment call.

E. RADIATION INTERCHANGE SEGMENTS

The computation and output of radiation interchange data involves the use of two program segments. Segment GBCAL utilizes the matrix of form factors for the problem, together with the surface properties, to compute a like-sized matrix of radiant interchange factors. This matrix is used in turn by segment RKCAL. RKCAL has the primary function of providing radiation interchange data in a BCD format that can be directly input to a thermal analyzer such as SINDA. RKCAL also performs the function of combining nodes used by TRASYS into larger nodes used by the thermal analyzer. This is done according to instructions provided by the user in the correspondence data block.

1. Segment GBCAL

Computation of grey-body radiation interchange factors involves the inversion of a matrix of the order of the problem size (number of nodes). TRASYS predecessor programs accomplished this using a scheme that stored entire matrix in core and required a large additional amount of core for working storage. This effectively limited thermal radiation problem size to around 140 nodes using the 64,000 word computers in current use. This was overcome to some extent in an option available in MTRAP 1 that inverted order 600 matrices using an iterative technique. This, however consumed a large amount of central processor time.

Since the TRASYS contract statement of work specified a 1000 node problem size, it was necessary to develop a new matrix inversion technique. This was done for program increment 1 by adapting an algorithm currently used in structural dynamics analysis. This method involves

using Cholesky's method to factor the form factor matrix into an upper and lower triangular matrix, each of which is the transpose of the other. From this point, fairly straightforward matrix algebra is used to invert the matrix and make the necessary additional passes required to compose the grey-body matrix. The core requirement for this method is only 2 times the problem size, plus storage space for the code.

2. Segment RKCAL

Segment RKCAL was developed using similar methods and some of the code from the like portion of MTRAP 1. The radiation conductors are printed, punched and/or written to tape in a small subroutine that may be easily modified by the user through the subroutine data block. This provides the user the capability to output his radiation conductors in a format compatible with any thermal analyzer program.

F. INCIDENT AND ABSORBED RADIATION SEGMENTS

In program increment 1, the computation and output of absorbed environmental heat rates involved up to four program segments. Segment DICAL is used to compute the direct incident heat fluxes, in the solar and infrared wavebands. This may involve the use of segment SFCAL to compute and/or load into core the shadow factor tables used in the flux shading computations. Segment AQCAL uses previously computed direct flux data, together with two grey-body factor matrices (one solar, one infrared) to compute the energy absorbed by each node at a point in orbit. This energy consists of four components: absorbed incident solar, absorbed incident infrared, absorbed reflected solar, and absorbed reflected infrared.

Segment QOCAL is analogous to segment RKCAL in that it writes out data in thermal analyzer input format, and the user's correspondence data is accounted for. The output consists of tables of energy vs time for each thermal analyzer node. An alternative form of output is orbital average heat loads for each node.

1. Segment DICAL

At the beginning of the program increment 1 development period, the computation routines for segment DICAL were in checkout as part of the MTRAP 2 development. This effort was continued and new code was developed to satisfy statement of work requirements. New code was required to provide the following:

- a. Orbit definition using classic orbit mechanics parameters.
- b. Simplified orbit definition.
- c. Heat source location from look angles.
- d. Star orientation.
- e. Shadowing from shadow factor tapes.

This code was generated and integrated into the DICAL segment. The checkout process was quite involved because the computation routines used as a foundation were not operational and contained errors in the planet flux calculation logic.

In common with the form factor computation segment, DICAL computation methods depart from previous practice in MTRAP 1 and LOHARP in the automatic elemental grid sizing according to user-input accuracy criteria. This logic is explained in the following two sections.

2. Direct Incident Solar Flux Accuracy

- a. Elemental Grid Sizing with No Shadowing

The direct solar flux incident upon the finite area A_I (Fig. V-3) is defined as

$$QDS_I = \frac{Q_S}{A_I} \int_{A_I} \cos \theta_1 dA_I \quad (V-7)$$

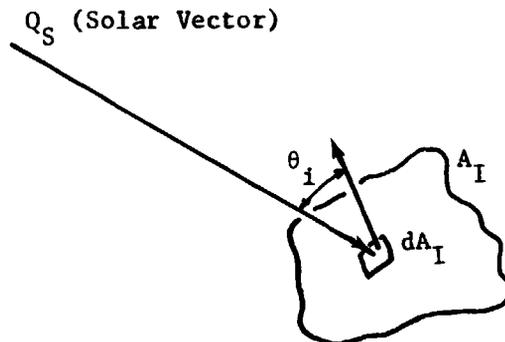


Figure V-3. Determination of Direct Incident Solar Flux

A finite difference approximation of Equation V-7 is

$$QDS_I = \frac{Q_S}{A_I} \sum_{i=1}^n A_i \cos \theta_i \quad (V-8)$$

Equation V-8 approaches an exact solution as the size of the elemental area, A_i , approaches zero.

To calculate the direct incident solar flux on an unshadowed node, the TRASYS program uses Equation V-8 with the number of elements, n , arbitrarily set to 10 to obtain a representative mapping of the node. Some run time optimization may result if n were set to 1 for planar nodes.

b. Elemental Grid Sizing Considering Shadowing

In the detailed shadowing checks, an element is either completely shadowed from the sun or not at all. The accuracy of representing the shadow, therefore, is proportional to the number of elements on the node. Since shadowing becomes increasingly important as the form factor from the node to the sun becomes larger, the number of elements required for accurate shadowing is considered to be proportional to the unshadowed node-to-sun form factor. That is,

$$N_{S_I} = \frac{QDS_I}{Q_S} \frac{B}{DIACCS} \quad (V-9)$$

where:

QDS_I is the unshadowed solar flux from Equation V-8 with n arbitrarily set to 10.

Q_S is the solar constant.

$\frac{QDS_I}{Q_S}$ is the unshadowed node-to-sun form factor.

B is a constant of proportionality (assigned a value of 25 in TRASYS).

DIACCS is a user-defined shadowing accuracy factor (defaults to 0.1 if not input).

The number of elements used on any given node for direct incident solar flux calculations is as defined by Equation V-9. However, values greater than 100 or less than 10 are not allowed.

3. Direct Incident Albedo and Planetary Fluxes

a. Elemental Grid Sizing with No Shadowing

The form factor from a finite nodal area, A_I , to the visible planetary area, A_p , (Fig. V-4) is defined as

$$F_{IP} = \frac{1}{A_I} \int_{A_I} \int_{A_p} \frac{\cos \theta_i \cos \theta_p}{\pi r_{ip}^2} dA_p dA_p \quad (V-10)$$

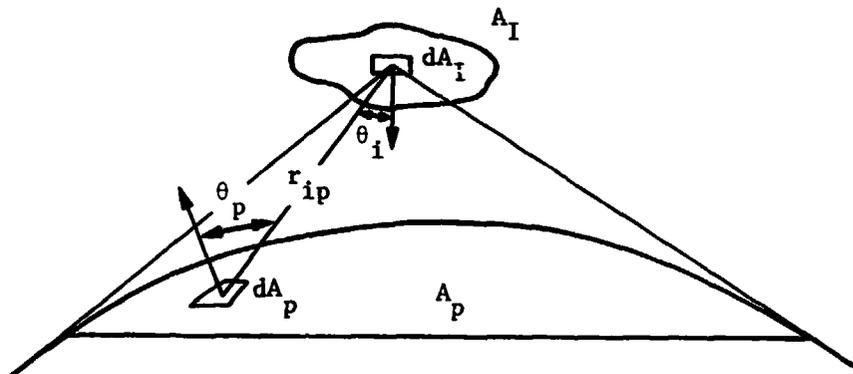


Figure V-4 Determination of Node-to-Planet Form Factors

A finite difference approximation of Equation V-10 is

$$F_{IP} = \frac{1}{A_I} \sum_{i=1}^n \sum_{p=1}^m \frac{\cos \theta_i \cos \theta_p}{\pi r_{ip}^2} A_i A_p \quad (V-11)$$

Equation V-11 approaches an exact solution as the sizes of the elemental areas, A_i and A_p , approach zero.

In TRASYS, the visible planet surface is divided into three zones as shown in Figure V-5. Fifty percent of the planet nodes are allocated to A_1 , thirty-seven percent to A_2 , and the remainder to A_3 so that higher concentrations of elements exist in the zones that have the most influence on spacecraft nodes.

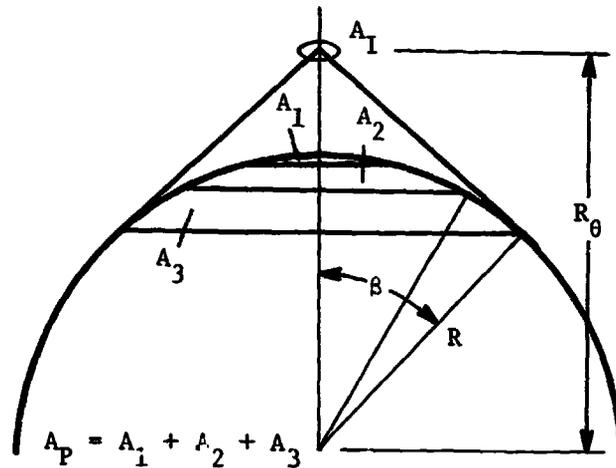


Figure V-5 Planet Element Breakdown

In order to control the accuracy of node-to-planet form factors, it is convenient to define the dimensionless ratio

$$DIACC = \frac{A_p \cos \theta_i \cos \theta_p}{r_{ip}^2} \quad (V-12)$$

The TRASYS program employs a technique using Equation V-12 to automatically establish the number of planet elements consistent with the user-defined accuracy parameter, DIACC. (DIACC defaults to 0.25 if it is not defined by the user.) TRASYS uses a simple arithmetic average of the numbers of elements indicated by the various node-planet element pairs. That is,

$$N_{OPT_{PI}} = \frac{A_p}{DIACC m_i m_p} \sum_{i=1}^{m_i} \sum_{p=1}^{m_p} \frac{\cos \theta_i \cos \theta_p}{r_{ip}^2} \quad (V-13)$$

where m_i and m_p are initial numbers of elements arbitrarily assigned to obtain a representative mapping of node I and the visible planet area.

In TRASYS the number of nodal elements, m_i , is set to 8. The initial number of planet elements is established by considering m_p to be directly proportional to the planet radius, R , and indirectly proportional to the orbit radius, R_θ . That is,

$$m_p = 25 \frac{R}{R_\theta} \quad (V-14)$$

Fifty percent of the total number of elements defined by Equation V-13 are uniformly distributed over A_1 (see Figure V-5), thirty-seven percent over A_2 , and the remainder over A_3 .

b. Elemental Grid Sizing with Shadowing

In the detailed shadowing checks, a nodal element is either completely shadowed from the planet element or not at all. The accuracy of representing the shadow, therefore, is proportional to the number of elements on the node. Since shadowing becomes increasingly important as the form factor from the node to the planet becomes larger, the number of nodal elements required for accurate shadowing is considered to be proportional to the unshadowed node-to-planet form factor. That is,

$$N_{S_{IP}} = \frac{B F_{IP}}{DIACCS} \quad (V-15)$$

where

F_{IP} is the unshadowed node-to-planet form factor.

B is a constant of proportionality (assigned a value of 3.0 in TRASYS).

DIACCS is a user-defined shadowing accuracy factor (defaults to 0.1 if not input).

The number of nodal elements is limited to the range $1 \leq N_I \leq 25$ for flat surfaces and $5 \leq N_I \leq 25$ for surfaces of revolution. The number of planetary elements is limited as follows:

$$A_1: 3 \leq N_{P_1} \leq 200$$

$$A_2: 2 \leq N_{P_2} \leq 148$$

$$A_3: 1 \leq N_{P_2} \leq 52$$

or

$$6 \leq N_P \leq 400 \text{ (total).}$$

4. Segment SFCAL

The primary function of SFCAL is to compute tables of shadow factors for use in the computation of direct irradiation by DICAL. Its primary output mode is to magnetic tape, though the tables are also printed as they are computed, and an image of the magnetic tape is also placed in out-of-core disc/drum storage to be used if a DICAL execution is encountered prior to the end of the job. If SFCAL is executed with a shadow factor tape mounted as a read tape, it does nothing more than transfer the tables to disc/drum storage for use by DICAL. This type of operation also provides a very convenient restart from a shadow tape that is partially complete due to interruption of a previous SFCAL run. A shadow data block in the input stream allows the user to input or override any shadow table values desired.

Shadow factor tapes contain one shadow factor table per node in the problem. These tables consist of one shadow factor for each of 135 vectors distributed in an omnidirectional manner from the problem's central coordinate system origin.¹ A shadow factor is defined as the percentage of nodal area, projected normally to one of the

¹ *Thermal Radiation Analysis System, (TRASYS), User's Manual.*
Martin Marietta Corporation, May 1973. Appendix C.

"energy source" vectors that would be illuminated by a collimated light beam collinear with the vector. Thus, nodes with no "view" of other parts of the problem geometry have shadow factors everywhere equal to 1.0 except for vectors below their "horizon," where the shadow factors are zero.

The DICAL routines that compute shadowing for solar flux calculations are quite applicable to computing shadow factor tables. Thus, the development of SFCAL did not require generating a new code for the entire segment.

5. Segment AQCAL

Since AQCAL functions only to compute the reflected component of absorbed heat rates and convert direct fluxes into direct absorbed heat rates, it is not strictly necessary that it be separate from DICAL. From a practical standpoint, however, AQCAL results in a valuable saving of compute time. For large problems, a series of restarts may be performed before direct irradiation is computed for all points in orbit. If absorbed heats had been computed after each orbit point, the time for the absorbed heat as well as grey body matrix calculations would be lost for each restart run. This is because a complete set of heat tables must be in storage before heat vs time tables can be output. Thus, the recommended procedure is to delay computation of the grey-body matrix and the absorbed heats until a complete set of direct flux tables are computed. AQCAL and QOCAL make this procedure convenient to the user, at his option.

6. Segment QOCAL

Segment QOCAL has the function of writing absorbed heat rate vs time tables and/or orbital average heat rates out to cards or tape. This data is in a BCD format identical with the SINDA thermal analyzer program's input. The user may print this data in any format desired by loading his own version of subroutine QOTABS, the QOCAL routine that writes out the data. The QOCAL segment used in program increment 1 was developed using a code that performed the same function in MTRAP 1. An ordering process is used to provide time tables in ascending order and a trapezoidal rule averaging routine is used for average heat rates. If node combining instructions are in the correspondence data block, the heat rate tables will reflect this.

VI. PROGRAM INCREMENT 2 DEVELOPMENT

Program increment 2 was developed over the time period from May 1973 through August 1973. This increment included the incorporation of specular-diffuse radiation interchange capability, output data plotting capability and capability to use trajectory tapes for defining spacecraft-heat source geometric relationships.

A. SPECULAR-DIFFUSE RADIATION INTERCHANGE

The specular-diffuse radiation interchange segment RBCAL was developed by incorporating the code developed under a concurrent benefit IR&D into the TRASYS structure.

The RBCAL segment is called from the FFCAL segment if specular-diffuse surfaces are present in the problem. RBCAL then constructs images of all the nodes "seen" by other nodes in the specular surfaces. These images are passed back to the form factor computation logic as any other node and the modified form factors to them are computed. They are modified in the sense that the specular reflectivity of the specular-diffuse surface involved is accounted for. This results in an expanded form factor matrix which is then compressed back to the ordinary size and the form factors stored for use by GBCAL. GBCAL computes radiant interchange factors that account for the specular reflectivities of specular-diffuse surfaces. Also, the user need not concern himself with specular-diffuse surfaces beyond defining the properties in his surface data. Specular-diffuse surfaces must be planar. This restriction exists because no logic has been developed to construct "carnival mirror" images that would result from curved surfaces.

1. Specular-Diffuse Radiation Theory

a. Nomenclature

A_i	= area of surface i
D	= coefficient matrix in absorption factor equation
F_{ij}	= form factor from surface i to surface j
$F_{ij(k)}$	= form factor from surface i to the image of surface j as seen in surface k
\mathcal{F}_{ij}	= radiation interchange factor coupling surface i and surface j
N	= total number of surfaces in radiation enclosure
$q_{i,net}$	= net heat flux leaving surface i
T_i	= temperature of surface i

- β_{ij} = absorption factor representing that fraction of energy leaving surface i that is finally absorbed by surface j
- δ_{ij} = Kronecker's delta
($\delta_{ij} = 1, i = j; \delta_{ij} = 0, i \neq j$)
- ϵ_i = hemispherical emissivity of surface i
- ρ_{ij} = hemispherical-directional reflectance of surface i in the direction of surface j
- ρ_{d_i}, ρ_{s_i} = diffuse and specular components of reflectance, respectively, for surface i
- σ = Stefan-Boltzmann constant
- τ_i = transmissivity of surface i
- ϕ_{ij} = image factor representing the fraction of energy leaving surface i and arriving at surface j both directly and by all possible intervening first-order specular reflections

b. Analytical Development

The following assumptions and ground rules were used in the analytical development presented herein for specular-diffuse radiation analysis techniques.

- 1) All surfaces are considered to be semigray (accounts for absorption and reflection, but no emission in the ultraviolet portion of the spectrum; accounts for absorption and reflection as well as emission in the infrared portion of the spectrum).
- 2) Equations will be developed for use in analyzing radiation enclosures consisting of diffuse, specular, and/or diffuse-plus-specular surfaces using an imaging technique.
- 3) All surfaces will be considered to emit diffusely and to reflect with diffuse and specular components such that the relationship

$$\epsilon_i + \rho_i^d + \rho_i^s + \tau_i = 1$$

is satisfied.

- 4) All surfaces with specular components of reflectance are restricted to planar surfaces to simplify imaging.
- 5) Only first-order images are considered (that is, no images of images or images in images will be generated).

The development of equations for radiation interchange factors follows the same procedure for both the infrared and ultraviolet portions of the spectrum with the only differences being in notation. Therefore, only those equations applicable to the infrared portion of the spectrum are developed here.

Consider a radiation enclosure consisting of N surfaces. The net heat flux from any one of these surfaces can be represented by

$$q_{i,\text{net}} = \sigma \sum_{j=1}^N \mathcal{F}_{ij} (T_i^4 - T_j^4)$$

where \mathcal{F}_{ij} is the radiation interchange factor that couples surface i to surface j.

The method of approach that is applied here in the development of radiation interchange factor (\mathcal{F}_{ij}) equations is an extension of the method set forth by Gebhart¹ for purely diffuse enclosures. The special utility in this formulation is that it yields coefficients that represent the fraction of energy emitted by a surface that is absorbed by another surface after reaching the absorbing surface by all possible paths.

Considering first-order images only, the general equation for the Gebhart type absorption factors for a diffuse-plus-specular enclosure can be written.

¹Gebhart, B. *Unified Treatment for Thermal Radiation Transfer Processes--Gray, Diffuse Radiators and Absorbers*, Paper No. 57-A-34, ASME, December 1957.

$$\beta_{ij} = \epsilon_j F_{ij} + \epsilon_j \sum_{k=1}^N \rho_k^s F_{ij(k)} + \sum_{m=1}^N \rho_m^d F_{im} \beta_{mj} + \sum_{k=1}^N \sum_{m=1}^N \rho_m^d \rho_k^s F_{im(k)} \beta_{mj};$$

$$i=1, 2, \dots, N; j=1, 2, \dots, N \quad (\text{VI-2})$$

By means of a term by term examination, Equation VI-2 can be interpreted as follows:

The fraction of the energy leaving surface i that is finally absorbed by surface j equals the sum of:

- 1) the energy that goes directly from surface i to j and is absorbed,
- 2) the energy that goes from surface i to surface j by all possible first-order specular reflections and is absorbed,
- 3) that fraction of the energy that goes directly from surface i to each of the surfaces in the enclosure, finally arrives at surface j by all possible paths, and is absorbed, and
- 4) that fraction of the energy that goes from surface i to each of the surfaces in the enclosure by all possible first-order specular reflections, thence to surface j by all possible paths, and is absorbed.

Rearranging the terms in Equation VI-2, yields

$$\beta_{ij} = \epsilon_j \left[F_{ij} + \sum_{k=1}^N \rho_k^s F_{ij(k)} \right] + \sum_{m=1}^N \rho_m^d \left[F_{im} + \sum_{k=1}^N \rho_k^s F_{im(k)} \right] \beta_{mj};$$

$$i=1, 2, \dots, N; j=1, 2, \dots, N \quad (\text{VI-3})$$

Equation VI-3 can be further simplified by defining an "image factor" (ϕ_{ij}) as that fraction of the energy that leave surface i and arrives at surface j both directly and by all possible first-order specular reflections, so that

$$\zeta_{ij} = F_{ij} + \sum_{k=1}^N \rho_k^s F_{ij}(k);$$

$$i=1, 2, \dots, N; k=1, 2, \dots, N. \quad (\text{VI-4})$$

Substitution of Equation VI-4 into Equation VI-3 yields

$$\beta_{ij} = \epsilon_j \phi_{ij} + \sum_{m=1}^N \rho_m^d \phi_{im} \beta_{mj}. \quad (\text{VI-5})$$

Rearrangement of the terms in equation VI-5 yields

$$\sum_{m=1}^N (\delta_{im} - \rho_m^d \phi_{im}) \beta_{mj} = \epsilon_j \phi_{ij}, \quad (\text{VI-6})$$

Equation VI-6 can be represented in matrix form as

$$\begin{bmatrix} D_{ij} \end{bmatrix} \begin{bmatrix} \beta_{ij} \end{bmatrix} = \begin{bmatrix} \epsilon_j \phi_{ij} \end{bmatrix} \quad (\text{VI-7})$$

where D is an $N \times N$ coefficient matrix with a general element

$$D_{ij} = \delta_{ij} - \rho_j^d \phi_{ij}. \quad (\text{VI-8})$$

The systems of equations represented by VI-7 can be solved by matrix inversion to obtain the absorption factors (β_{ij})

$$\beta_{ij} = \sum_{m=1}^N D_{im}^{-1} \epsilon_j \phi_{mj}. \quad (\text{VI-9})$$

The radiation interchange factors (\mathcal{F}_{ij}) are related to absorption factors by the expression

$$\mathcal{F}_{ij} = \epsilon_i \beta_{ij}. \quad (\text{VI-10})$$

and, using the usual arguments for the conservation of energy, the reciprocity relation for the N^2 values of F_{ij} is

$$A_i F_{ij} = A_j F_{ji} \quad (\text{VI-11})$$

The foregoing equations apply to radiation enclosures consisting of any combination of diffuse and specular surfaces ranging from totally diffuse to totally specular enclosures.

B. OUTPUT DATA PLOTTER SEGMENT

The output data plotter segment is a generalized plotting package set up to retrieve data from out of core storage files and generate X vs Y plots. Convenient input instruction is provided to obtain the following types of plots:

- absorbed or incident heat fluxes;
- absorbed or incident heat rates.

The choice of solar, albedo, or planetary is available or the sum of all. Also, the individual solar, albedo, and planetary curves may be plotted along with the total on the same frames. Plots of any other data may be obtained by writing the logic to locate said data in plotter accessible storage files. The data plotter uses a sophisticated curve fit routine using the method of tangents. Logic is provided to handle the discontinuities common in orbital heat data. A straight line curve fit is an option. The data plotter segment was developed based on the plot capability of MTRAP 1. New logic was written to provide generality and convenient plot control input.

C. TRAJECTORY TAPE CAPABILITY

Capability to use a trajectory tape to define the spacecraft-planet-sun geometric relationship was provided by writing two subroutines, DITTP and DITTPS that are called directly by the user. Since no standard exists for trajectory tapes, this subroutine was written so that the user could, in effect, define his trajectory tape format with an initial call to DITTP. At subsequent points in the trajectory, calls to DITTPS are made with only mission time as an argument. The only limitation to the approach used is that the trajectory tape must define the sun and planet locations in terms of look angles.

VII. PROGRAM INCREMENT 3 DEVELOPMENT

Program increment 3 was developed over the period from October 1973 through March 1974. This increment consisted of a series of improvements in input and output capability, together with a major conversion effort to allow TRASYS to run on IBM 370 series operating systems.

A. MODIFICATION OF LARGE BLOCKS OF SURFACE DATA

Running large TRASYS models identified the need for a convenient means to change the unit of length used in blocks of surface data. Also, a convenient means to change the model surface identification numbers would ease the labor of merging independently developed TRASYS models together.

These functions were provided by writing preprocessor routines that recognize special cards in the surface data block and process the surface data accordingly. When a D-card is encountered in the surface data, all linear dimension data and values found on the surface data cards following are multiplied by a factor found on the D-card. All surfaces will be thus modified until another D-card, with another multiplier, is encountered. When an N-card is encountered, all surface and node numbers on the cards following are changed by a positive or negative integer increment found on the N-card. This process is terminated or changed by another N-card.

B. ADDED SURFACE DATA INPUT OPTIONS

Preprocessor code was written to recognize two additional types of surfaces. Both types relate to surfaces previously defined in the surface data, and are basically a shorthand means of surface definition. When a DUP surface is encountered, a surface identical to a previously input surface is constructed, with exceptions defined by the user. The primary use of DUP surfaces is to allow identical parts of a configuration to be defined but once, and then located as desired using the DUP option.

IMAGE surfaces are similar to DUP surfaces except that they are mirror images of a previously defined surface. Any characteristics of the imaged surface may be redefined for the image except geometric shape and size. The imaging plane is defined using three Cartesian coordinate points input as a special surface.

C. AUTOMATIC NODE PLOTS

At computer installations where an interactive graphics mode on remote terminals is not available, the time required to obtain plot output is sometimes excessive. This led to a need to provide pictorial plots of problem geometry even if surface data input errors did not allow a plot of all the input surfaces.

This capability was provided by writing preprocessor code that processes and stored unflawed surface definitions at the user's option, for an automatic execution of the NPLOTT processor segment. This required a rather extensive change because a basic premise of TRASYS design was that a complete and error free preprocessor execution must precede any processor segment executions.

When the error plot option is activated, and surface errors are encountered, the user's operations data input is voided and a substitute operations data block is written by the preprocessor. This operations data provides for four automatically scaled node plots of each subblock of the surface data.

D. AUTOMATIC ORBIT GENERATION

The specialized operations data block language used to direct TRASYS problem solutions provides the generalized capability necessary to cope with variable geometry and reorientation in orbit. It is, however, somewhat tedious to write the logic to step through 12 or more points in orbit, especially when the planet shadow points are not known.

The automatic orbit generation, or ORBGEN option was generated to eliminate this problem. The ORBGEN function is supplied by 4 preprocessor routines that write extensive blocks of operations data block code and pass it on with the remainder of the user's operations data block. When a single card containing initial and final true anomaly values, number of points desired and a mnemonic identifying the orbit type is encountered, the ORBGEN routines write the necessary code. Four orbit options are allowed: inertial (sun or star) oriented for any orbit, planet oriented, planet oriented in circular orbit, and no planet (heliocentric).

E. SHADOW ONLY SURFACES

A shadow-only surface is one that is so far from the region

of thermal interest that the analyst has no interest in its temperature, per se, but desires only that its effects on other parts of the problem be accounted for.

This function was provided by allowing ONLY as an option for the SHADE/BSHADE flag in surface definition. Surfaces identified as such are added to the end of the node array and provide for blockage only in the form factor and direct irradiation links. They appear nowhere in the program output. The energy conservation aspects of shadower-only surfaces are provided for as follows:

1. Form factors from a general node i to a shadower-only surface are summed and added to F_{ii} . This infers that $T_{\text{shadowers}} = T_i$.
2. Incoming irradiation is blocked by shadower-only surfaces. Outgoing radiation is also blocked because these surfaces reduce the form factor to space of the general node i . This too infers that $T_{\text{shadowers}} = T_i$.

F. RADIATION NETWORK SIMPLIFICATION

The number of radiation conductors that can result from a 1000 node enclosure can approach 500,000. This is far in excess of the capacity of current thermal analyzer programs both from the standpoint of compute memory and the resulting run time. A program segment was developed to allow the user to control this problem with two methods that provide an approximation to the radiation interchange of large enclosures.

The first of these methods, referred to as the Multiple Enclosure Simplification Shield (MESS) technique,^{1,2} allows a large radiation enclosure to be modularized into discrete subenclosures by the assignment of imaginary interface shield nodes. Each of these smaller nodes can be analyzed independently of the others.

The second method, referred to as the Effective Radiation Node (EPN) technique, is used to reduce the number of radiation conductors required to thermally model an enclosure

¹Chapter, J. J. and Connolly, J. M., "Simplified Radiation Analysis Using Modularized Enclosures," *Journal of Spacecraft and Rockets*, Vol. 8, No. 9, September 1971, pp. 1006-1008.

²Chapter, J. J. and Connolly, J. M., "Radiation Condenser-User's Manual," Technical Memo 0478-71-16. Martin Marietta Corporation, Denver, Colorado, April 1971.

(or subenclosure) by replacing small conductors from each r. with a single conductor coupled to the enclosure's ERN.

1. Radiation Condensor Theory

In this section, the symbol F denotes the general grey body radiant interchange factor. The Multiple Enclosure Simplification Shield (MESS) technique and the Effective Radiation Node (ERN) technique are independent and can be discussed separately. Consider an N-node radiative enclosure that forms a section of a complex thermal model. The temperature of node i is a function of thermal radiation coupling and the applied head load, Q_i (assume that heat loads resulting from conduction and convection are included in Q_i). The steady-state temperature of node i is then given by

$$T_i = \left[\left(\sum_{j=1}^N \sigma A_i F_{ij} T_j^4 + Q_i \right) / \sum_{j=1}^N \sigma A_i F_{ij} \right]^{1/4} \quad (\text{VII-1})$$

a. ERN Technique

In applying the ERN technique, the enclosure radiation conductors for the ith node are divided into P_i primary and $N-P_i$ secondary couplings. The summation term in the numerator of Equation (VII-1) can then be written as follows:

$$\sum_{j=1}^N \sigma A_i F_{ij} T_j^4 = \sum_{k=1}^{P_i} \sigma A_i F_{ik} T_k^4 + \sum_{\ell=P_i+1}^N \sigma A_i F_{i\ell} T_\ell^4 \quad (\text{VII-2})$$

The number of radiation conductors can be reduced by arranging the conductors in decreasing order of the numeric value ($A_i F_{ij}$) and replacing the secondary coupling summation of Equation (VII-2) with a single conductor coupled to an ERN. That is,

$$\sum_{\ell=P_i+1}^N \sigma A_i F_{i\ell} T_\ell^4 = \left[\sum_{\ell=P_i+1}^N \sigma A_i F_{i\ell} \right] T_{\text{ERN}}^4 \quad (\text{VII-3})$$

The ERN temperature is calculated by the thermal analyzer program as a steady-state node temperature based on a fourth-power conductor weighted average of the enclosure node temperature using the secondary conductors.

$$T_{ERN} = \left[\frac{\sum_{i=1}^N \sum_{l=P_i+1}^N \sigma_{A_i F_{il}} T_i^4}{\sum_{i=1}^N \sum_{l=P_i+1}^N \sigma_{A_i F_{il}}} \right]^{1/4}. \quad (\text{VII-4})$$

Using the relationships of Equations (VII-2 and (VII-3), the approximate *i*th node temperature can be written from Equation (VII-1) as a function of the ERN temperature.

$$T_i = \left\{ \frac{\left[\sum_{k=1}^{P_i} \sigma_{A_i F_{ik}} T_k^4 + \left(\sum_{l=P_i+1}^N \sigma_{A_i F_{il}} \right) T_{ERN}^4 + Q_i \right]}{\sum_{j=1}^N \sigma_{A_i F_{ij}}} \right\}^{1/4} \quad (\text{VII-5})$$

b. Application of the ERN technique

The significant radiation fraction defined by the relationship

$$\text{RFRAC} = \frac{\sum_{k=1}^{P_i} \sigma_{A_i F_{ik}}}{\sum_{j=1}^N \sigma_{A_i F_{ij}}} \quad (\text{VII-6})$$

is specified by the user. The number of primary conductors, P_i , is determined by summing conductor values

for a given node until the sum is greater than the fraction RFRAC of the sum of all conductors to the node. That is,

$$\sum_{k=1}^{P_i} \sigma_{A_i F_{ik}} > \text{RFRAC}^* \sum_{j=1}^N \sigma_{A_i F_{ij}} \quad (\text{VII-7})$$

All primary and reverse direction conductors are flagged to be used intact. The secondary conductors for each node are summed to determine the conductor value for the node-to-ERN coupling.

Since the error in the approximate temperature is a function of the enclosure temperature band, the ERN technique results can be improved if nodes that deviate significantly from the average temperature of the enclosure are not coupled to the ERN. These analyst-defined nodes are referred to as special nodes.

The percentage reduction in enclosure conductors and subsequent network error as a result of applying the ERN technique is controlled by the analysts selection of an RFRAC value consistent with known accuracy of problem parameters (enclosure geometry, surface optical properties, etc).

Experience has shown that the greatest percentage reduction in conductors results for enclosures with more than 75 nodes, significant shadowing and low emittance surfaces. An RFRAC value of 0.7 has been found to result in a significant reduction in conductors with acceptable error for typical radiation enclosures.

c. MESS Technique

The MESS technique provides the analyst with a means of dividing a radiation enclosure into an arbitrary number of subenclosures. MESS node pairs are defined by the analyst at the interface between subenclosures as two planar surfaces with the property of absorbing and emitting all energy incident upon them (black surfaces). Consider an N-node subenclosure, n, as shown in Figure VII-1 where subscripts r and r' refer to the MESS node pair of the nth and jth subenclosures, respectively. Temperatures in n are affected by T_{MESS_r} , that repre-

sents the average thermal effect of the j subenclosure nodes on the nodes of n. The primary conductors of Equation (VII-2) include conductors to MESS nodes. For a general subenclosure, n, with R_n interface MESS nodes, the primary radiation coupling summation for node i is

$$\sum_{k=1}^{P_i} \sigma A_i F_{ik} T_k^4 = \sum_{r=1}^{R_n} \sigma A_i F_{ir} T_{MESS_r}^4 + \sum_{k=R_n+1}^{P_i} \sigma A_i F_{ik} T_k^4 \quad (VII-8)$$

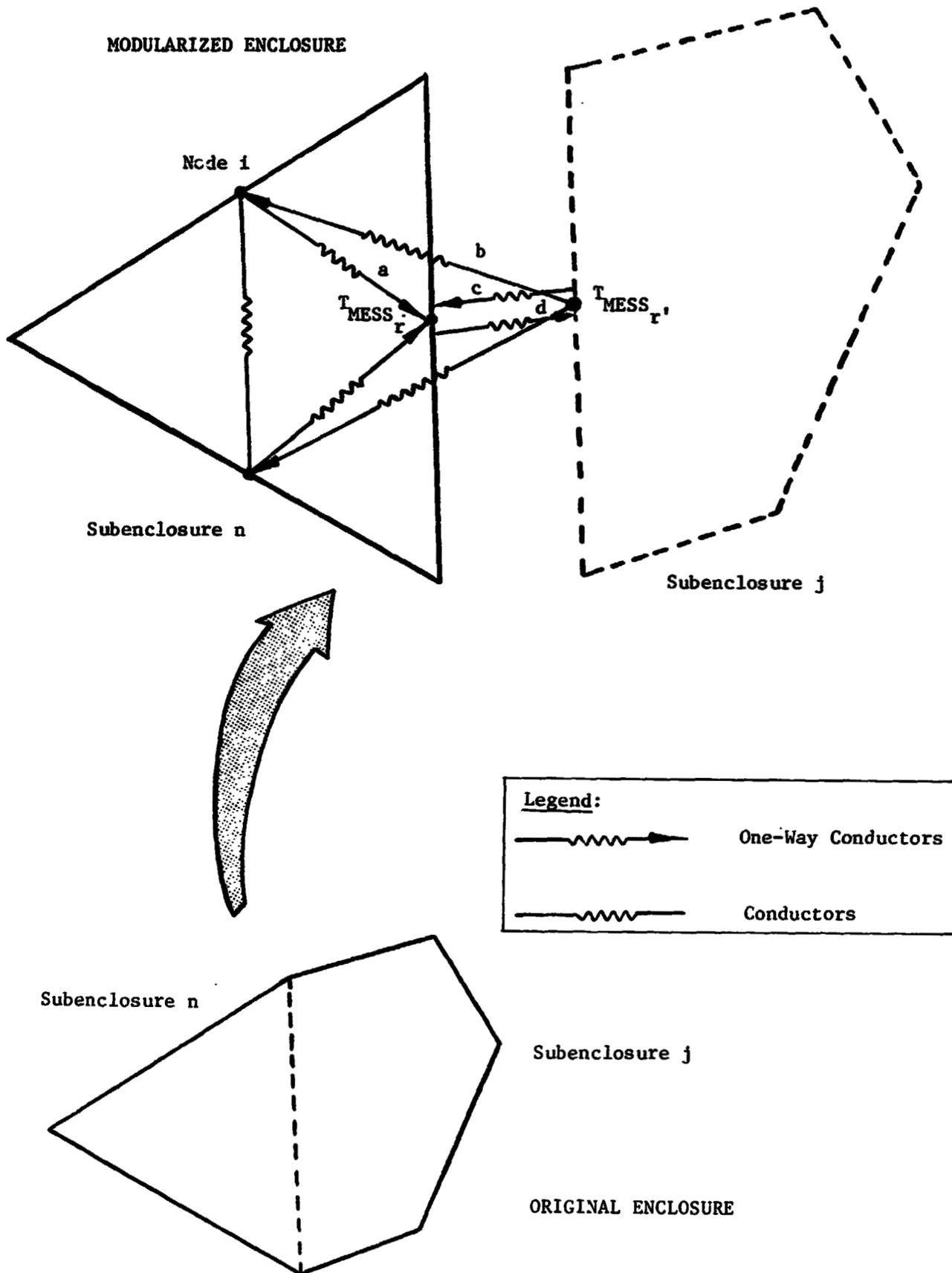


Figure VII-1 MESS Technique One-Way Conductors

An energy balance on MESS node r' gives

$$T_{\text{MESS } r'} = \left[\left(\sum_{\substack{m=1 \\ m \neq r'}}^{R_j} \sigma_{r',F_{r',m}} T_{\text{MESS } m'}^4 + \sum_{k=R_j+1}^{P_{r'}} \sigma_{r',F_{r',k}} T_k^4 + \sigma_{r',F_{r',r}} T_{\text{MESS } r'}^4 \right) / \left(\sum_{\substack{m=1 \\ m \neq r'}}^{R_j} \sigma_{r',F_{r',m}} + \sum_{k=R_j+1}^{P_{r'}} \sigma_{r',F_{r',k}} + \sigma_{r',F_{r',r}} \right) \right]^{1/2} \quad (\text{VII-9})$$

$F_{r',r}$ represents the reflections between n and j due to nonblack subenclosure surfaces and is obtained from the radiation interchange matrix for each subenclosure.

The approximate temperature of the i th node is obtained from Equation (VII-5) using Equation (VII-8) as

$$T_i = \left\{ \left[\sum_{r=1}^R \sigma_{i,F_{ir}} T_{\text{MESS } r'}^4 + \sum_{\ell=R_n+1}^{P_i} \sigma_{i,F_{i\ell}} T_\ell^4 + \left(\sum_{h=P_i+1}^N \sigma_{i,F_{ih}} \right) T_{\text{ERN}}^4 + Q_i \right] / \left(\sum_{r=1}^R \sigma_{i,F_{ir}} + \sum_{\ell=R_n+1}^{P_i} \sigma_{i,F_{i\ell}} + \sum_{h=P_i+1}^N \sigma_{i,F_{ih}} \right) \right\}^{1/2} \quad (\text{VII-10})$$

The error in T_i is a complex function of the percentage of ERN secondary conductors, temperature band of the subenclosure nodes, and the number of subenclosures. In a variety of problems studied, the error has been found to be negligible.

d. Application of the MESS Technique

Generation of MESS one-way conductors from the sub-enclosure radiant interchange matrix requires that the analyst specify the interface MESS node pairs. As node conductors are generated MESS nodes are flagged and appropriate one-way conductors are generated for use in the thermal analyzer program.

The location of MESS node pairs in an enclosure is influenced by:

- 1) the number of subenclosure surfaces;
- 2) geometric considerations;
- 3) expected thermal gradients; and
- 4) the number of analysts available to work on the enclosure.

Optimum reduction in form factors and conductors occurs in large enclosures divided so that the subenclosures contain approximately equal numbers of nodes. For enclosures divided into two approximately equal subenclosures, up to 50% reduction in the number of form factors and conductors can be expected.

e. ERN/MESS Application

The ERN and MESS techniques can be applied separately or simultaneously as the particular problem dictates. When they are applied simultaneously, an ERN is defined for each subenclosure and the MESS nodes are considered to be special nodes, that is, MESS nodes are not coupled to the ERN.

G. EQUIVALENT FORM FACTORS

Many radiation enclosures involve geometry that is symmetric in some manner and many, therefore, have many form factors that are exactly equivalent to other form factors because the node pairs involved are the same size and shape and "see" each other in the same way. The analyst can identify these situations, and if he can conveniently enter this information, a considerable amount of computer time may be saved in form factor computation. This capability was provided under program increment 3 by a modification to the data processing routine associated with the form factor data block, and a

modification to the form factor computation segment that recognizes and utilizes the equivalent form factor data. This required an additional disc/drum storage file in the processor. A random access technique was used for storage and retrieval so that equivalent form factor capability is handled very efficiently from a processing time standpoint

H. PROCESSOR LIBRARY IMPROVEMENTS

1. Approximate Radiant Interchange Factors

A user callable subroutine was written that computes grey body interchange factors according to the first order approximation:

$$F_{ij} = \text{PROPI} * \text{PROPJ} * F_{ij}$$

where:

PROPI and PROPJ are the diffuse surface properties, solar absorptivity, or infrared emissivity.

F_{ij} is the approximate radiant interchange factor between surfaces i and j.

F_{ij} is the form factor.

This equation accounts for the direct path between i and j and ignores the reflection paths involving the remaining nodes in the enclosure. It is, of course, exact for a black enclosure (PROPI = PROPJ = 1.0 for all i, j) and gives a reasonable approximation if the surface properties are all 0.8 or greater. A call to this subroutine in lieu of executing the grey body computation segment will generate the approximate grey body factors and store them in the same manner as GBCAL. There is a significant saving in processing time if the problem size is greater than 200 nodes.

2. Data Modification Routines

A series of user-callable subroutines were developed that enable the analyst to change certain types of data during the execution of his problem. The routines allow changes in surface data parameters, thus providing a convenient way to perform many types of parametric studies without the necessity of making multiple runs.

Deleting the data modification calls from the operations data block is a convenient and foolproof way to return the model to baseline status. The modification routines allow changes in the following node/surface properties:

- Area;
- Diffuse IR emissivity and/or solar absorptivity;
- Specular IR and solar reflectivity;
- IR and/or solar transmissivities;
- Shade/beshade flags.

All data modification calls use the node identification number and the new value desired as arguments. The area change routine has a special "all" option that results in all the active areas being altered by a common multiplier.

3. Thermal Analyzer Interface Tape

In order to take advantage of all the input options available for the SINDA thermal analyzer program, as well as any options that may be developed in the future, it was decided to provide a completely generalized output capability for TRASYS. New output routines were written that generate a binary output tape that contains the following data:

- Processed surface data;
- Form factors;
- Grey body factors;
- Direct fluxes;
- Absorbed heats;
- Correspondence data.

Using this tape, an interface program, or routines in SINDA itself, can be used to process the data in any way desired and pass it on for thermal analysis.

I. DIRECT IRRADIATION WITH SPECULAR-DIFFUSE SURFACES

During the development of program increment 3, the direct irradiation segment was modified so that the effects of the sun's image in the specular surfaces is accounted for in the computation of direct fluxes. This is an improvement over the approach used in program increment 2, where specular effects were accounted for only in the computation of absorbed heats.

1. Development of Incident Flux Equations

a. Nomenclature

F_{is} = form factor from surface i to the sun,

$F_{is(k)}$ = form factor from surface i to the image of the sun in surface k ,

F_{si} = form factor from the sun to surface i ,

$F_{si(k)}$ = form factor from the image of the sun in surface k to surface i ,

q_i = incident flux on surface i ,

S = solar constant,

b. Incident Solar Flux

Considering first-order images only, the equation for the total solar energy incident upon i can be written

$$A_i q_i = S A_s F_{si} + S \sum_{k=i}^N \rho_k^s A_s F_{si(k)} \quad (\text{VII-11})$$

By reciprocity,

$$A_s F_{si} = A_i F_{is} \quad (\text{VII-12})$$

Also, observe that

$$A_s F_{si(k)} = A_s F_{s(k)i} \quad (\text{VII-13})$$

and, by reciprocity,

$$A_s F_{si(k)} = A_s F_{s(k)i} = A_i F_{is(k)} \quad (\text{VII-14})$$

Substituting Equations (VII-12) and (VII-14) into Equation (VII-11), dividing by A_i , and collecting terms, yields the desired equation for solar flux incident upon surface i .

$$q_i = S F_{is} + \sum_{k=1}^N k F_{is}^s(k) \quad (\text{VII-15})$$

Solution of Equation (VII-15) by TRASYS has the effect of increasing the incident solar flux on each of the surfaces that can "see" images of the sun in specular surfaces.

c. Incident Albedo and Planetary Fluxes

During the development of the specular-diffuse capability in the TRASYS program, it was decided by the contractor that the small improvement in accuracy to be gained by taking specular reflections of albedo and planetary fluxes into account did not justify the significant increase in computer run time.

For example, in the case of a sun-oriented vehicle orbiting a planet, the time to compute albedo and planetary fluxes was already much greater than that required to compute solar fluxes. The addition of one specular surface would approximately double this time, two specular surfaces would triple it, etc. Therefore, specular reflections of albedo and planetary fluxes have been ignored.

J. CONVERSION TO IBM 370

Conversion of TRASYS involved accomplishing the following major tasks:

1. Converting arrays containing Hollerith information of up to six characters per word into either doubly-dimensioned arrays or double precision variables in order to compensate for the four characters per word available on IBM. The use of shifting operations and/or packing of words made it impossible to do a mass double precision of all variables to compensate for this character manipulation problem.

The masking, shifting, packing, and unpacking of arrays and data all had to be modified to compensate for the difference in word size, number of characters per word and the octal/hexadecimal difference.

2. All disk and tape files had to be redefined and IBM job control language definitions had to be made for each individual file. Some files that were defined as both Binary and BCD had to be broken down into several individual files. The CDC and UNIVAC machines are able to use Hollerith descriptions with executable statements. All of these statements had to be converted through the use of DATA statements.
3. A routine had to be written that would allow the program to process data that was punched either in BCD or EBCDIC so as not to require the repunching of all data decks.
4. All multiple tape and disk files had to be redefined and pseudo end of files used. This was to allow the program to run without having to define all multifile data sets in the job control language stream.

VIII. PROGRAM EVALUATION AND OPTIMIZATION

Shortly after delivery of program increment 1 and the accompanying user's manual, TRASYS began to be used in the engineering thermal analysis environment. At that time, TRASYS consisted of approximately 30,000 FORTRAN cards, and, not surprisingly, program bugs began to turn up at a vigorous frequency.

It would serve no purpose to attempt to enumerate all the program "fixes" that were made. The majority were a varied mix of misspelled flags and program variables and minor logic errors. The one area that can be singled out as benefiting most from program evaluation was the routines that process Cartesian coordinate point input for surface definition. Fairly major rewrites were necessary in some of this code before the users could depend on their point input getting into the processor without problems.

Against this background of continued bug elimination, several areas were identified where optimization was definitely warranted. Some of these were placed in the statement of work governing program increment 3. Three additional areas where optimization was accomplished are described in the following sections.

1. Matrix Inversion

The original matrix inversion routine used in the grey-body factor segment was based on algorithms developed in an environment where there was no charge for the operations necessary to transfer data in and out of core memory. This changed at the Martin Marietta/Denver Computer Center in late 1973 and it suddenly became cost-prohibitive to invert the matrix associated with a 300-node problem then in work. The matrix inversion routine was rewritten using a blocking technique to do as many operations as possible in core between I/O operations. This allowed the 300 node problem to run in-core on the Martin Marietta 6500 computer, using six minutes of central processor time. This modification was added to the NASA-JSC Univac system before any large problems were attempted, so it is unknown to what extent the Univac version was improved.

2. Direct Irradiation Data Storage

Generation of absorbed heat vs time tables requires that absorbed heat data only be located in out-of-core storage for each orbit point. This was the initial design concept used. Program usage pointed out that this concept had limitations. When output data plotting capability was added, it was found that complete tables of fluxes were never generated unless the user wrote his operations

data specifically for data plots. This led to changes in the DICAL segment so that fluxes were always placed in storage. Another facet of this change was the generation of subroutine DICOMP, which provides the user the option to compute, obtain from previously stored data, or zero out the direct irradiation data for a given step. Besides eliminating the plot problem, this change provided a notable increase in operational flexibility.

3. Correspondence Data Storage

Large-model operations at JSC pointed out that the method used to store correspondence data for use by the RKCAL and QOCAL program segments was in effect the problem size limiter. This was eliminated by removing correspondence data from in-core storage to out-of-core storage in a random access file.

4. When polygons were provided as a surface type option, it was recognized that the user would never want to leave the polygons broken down into the individual triangles generated because of shadowing solution limitations. It was thought, however, that the user could provide the correspondence data necessary to recombine the occasional polygon he might use. This proved an unwarranted assumption, however. When "nonplanar" polygons (best described as a collection of congruent triangles) were made allowable, NASA-JSC users generated relatively large models that consisted almost entirely of nonplanar polygons. The correspondence data necessary to recombine polygons is now automatically generated by preprocessor routines and passed on with the user's correspondence data, thus, in effect, automatically combining the individual triangles back into polygons. However, the individual triangles still appear in the node plots.